[0328] Description ([Descr]). The optional description of a relationship can be used to help clarify to users the meaning of a relationship.

[0329] From Table ID ([Application_Table_ID]). Each relationship consists of (a) a join criteria relating tables, (b) a table that is the "source" of the join, (c) a table that is the "destination" of the join, and (d) intermediate tables that are required by the join. This column indicates the source table of the join.

[0330] Inverse Relate ID. Some relationships between tables should only be used in one direction, while others are bi-directional. For example, the road-to-county relationship "is in county" could also be used as a county-to-road relationship "contains roads". This column lists the relate ID of the relationship, if any, that is the inverse of the given relationship.

[0331] Relate Join Type. The type of join between tables is important in order when generating SQL statements to query the data. For example, a query to count accidents can simply use a Count() SQL command if all of the joins in the query are one-to-one, but must use a "group by" clause if not. The value of this column indicates the type of join created by a Relate.

[0332] Name ([Name]). Each entry in the Relates table has a name that is used to help users select a relationship of interest. Because this name will be used in pick lists to help users select related Entities that are part of a query, a name that will make sense in this context should be selected. For example, the relate between a road and the county that contains that road could be named "is in

county". The Name must be unique among the names that relate two specific Entity classes.

[0333] Relate ID ([App_Table_Relate_ID]). Each entry in the Relates table has a unique numeric ID, the Relate ID, that is used in to identify that relationship.

[0334] To Table ID ([Application_Table_ID_To]). Each relationship consists of (a) a join criteria relating tables, (b) a table that is the "source" of the join, (c) a table that is the "destination" of the join, and (d) intermediate tables that are required by the join. This column indicates the destination table of the join.

[0335] Where Clause ([Where_Clause]). Each relationship in the Relates table is defined by (a) the tables that form part of the relationship, which become part of the FROM clause of an SQL statement and (b) the join criteria relating the tables, which becomes part of the WHERE clause. This column contains the phrase that becomes part of the where clause.

[0336] The Relates table 1760 defines relationships between Application Tables 1740 via the Relates Tables table 1750. Each relationship consists of (a) a join criteria relating tables, (b) a table that is the "source" of the join, (c) a table that is the "destination" of the join, and (d) intermediate tables that are required by the join. This column indicates the destination table of the join. The first three criteria are specified by the Where Clause, From Table ID, and To Table ID columns of the Relates table. This information is sufficient to defined relationships that only involve the two primary tables. However, some relationships will involve some intermediate tables. For example, the relationship between the Division Section Entity and the

County Entity requires the Attribute Event table containing the county code as an intermediate table. The Relate Tables table 1750 lists the intermediate tables, if any, required to support a relationship, and includes the following columns:

[0337] Relate ID ([App_Table_Relate_ID]). Each entry in the Relate Tables table 1750 defines an intermediate table of a relationship. This column is the ID of the relationship.

[0338] Application Table ID ([Application_Table_ID]). Each entry in the Relate Tables table 1750 defines an intermediate table of a relationship. This column is the ID of the intermediate table.

[0339] Most Attribute values are directly related to a column in an Attribute table and the value in this column is entered and maintained directly. However, some Attribute values are derived from the values of other Attributes. For example, a Paved Attribute could be derived from a Pavement Type Attribute. The Derivations table 1770 provides the formula used to derive an Attribute value.

[0340] Attribute ID. Each derivation is used to generate the values of a derived Attribute. The Attribute ID is the ID of this derived Attribute and also serves as a unique ID for this derivation.

[0341] Derivation Type. A derived Attribute can either be derived on-the-fly during a query or instantiated as a column in an Attribute table. This column indicates which of these types of derived Attributes applies to this entry.

[0342] Description. This column contains an optional description of a derivation.